

# APPLICATION FOR UNITED STATES LETTERS PATENT

5  
10  
15  
20  
25  
30  
35  
40  
45  
50  
55  
60  
65  
70  
75  
80  
85  
90  
95  
100  
105  
110  
115  
120  
125  
130  
135  
140  
145  
150  
155  
160  
165  
170  
175  
180  
185  
190  
195  
200  
205  
210  
215  
220  
225  
230  
235  
240  
245  
250  
255  
260  
265  
270  
275  
280  
285  
290  
295  
300  
305  
310  
315  
320  
325  
330  
335  
340  
345  
350  
355  
360  
365  
370  
375  
380  
385  
390  
395  
400  
405  
410  
415  
420  
425  
430  
435  
440  
445  
450  
455  
460  
465  
470  
475  
480  
485  
490  
495  
500  
505  
510  
515  
520  
525  
530  
535  
540  
545  
550  
555  
560  
565  
570  
575  
580  
585  
590  
595  
600  
605  
610  
615  
620  
625  
630  
635  
640  
645  
650  
655  
660  
665  
670  
675  
680  
685  
690  
695  
700  
705  
710  
715  
720  
725  
730  
735  
740  
745  
750  
755  
760  
765  
770  
775  
780  
785  
790  
795  
800  
805  
810  
815  
820  
825  
830  
835  
840  
845  
850  
855  
860  
865  
870  
875  
880  
885  
890  
895  
900  
905  
910  
915  
920  
925  
930  
935  
940  
945  
950  
955  
960  
965  
970  
975  
980  
985  
990  
995

TITLE: CALL ADMISSION CONTROL

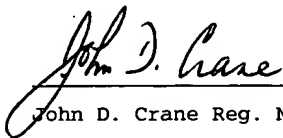
APPLICANTS: ALIREZA ABAYE, WING F.LO, and MOSES SUN

ASSIGNEE: Nortel Networks Limited  
World Trade Center of Montreal  
380 St. Antoine Street West, 8th floor  
Montreal, Quebec H2Y 3Y4, Canada

## CERTIFICATE OF EXPRESS MAILING

I hereby certify that this correspondence, which includes 52 pages of Specification (including 18 pages of claims), 1 page of Abstract and 12 sheets of Drawings, is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 C.F.R. Sec. 1.10 addressed to: Box Patent Application, Assistant Commissioner for Patents, Washington, D.C. 20231, on May 23, 2000

Express Mailing Label No.: EL158462914US

  
John D. Crane Reg. No. 25,231

This application is a continuation-in-part of U.S. patent application serial no. 09/370,984, filed August 20, 1999.

**FIELD OF THE INVENTION**

5

The invention relates to managing calls over a data network, such as an Internet Protocol (IP) network.

**BACKGROUND OF THE INVENTION**

10  
15

Packet-based data networks are widely used to link various nodes, such as personal computers, servers, gateways, and so forth. Packet-based data networks include private networks, such as local area networks (LANs) and Wide Area Networks (WANs), and public networks, such as the Internet. The increased availability of such data networks has increased communication among nodes, whether the nodes are located in close proximity to each other (such as within an organization) or at far distances from each other. Popular forms of communications across such data networks include electronic mail, file transfer, network printing, web browsing, and other exchanges of digital data.

With the increased capacity and reliability of data networks, voice communications over both private and public data networks have become possible. Voice communications in a conventional public switch telephone network (PSTN) provides users with dedicated end-to-end circuit connections for the duration of each call. Unlike PSTN network, voice communications over data networks, such as IP (Internet Protocol) networks, has to compete the network bandwidth with other non-voice data (e.g. electronic mail, file transfer, web access, and other

traffic). This makes the performance of voice data packets unpredictable if there is insufficient provision of quality of service.

In an IP network, each data packet is routed to a destination node based upon the destination IP address contained within the header of each packet. Each data packet may be  
5 routed over separate network paths before arriving at the final destination for processing.

Transmission speeds of the various packets may vary widely depending on the conditions of the data paths over which the data packets are transferred. During peak usage of data networks, significant delays may be added to the transfer of voice data packets causing poor quality of voice communications. If there is insufficient provision of QoS in a congested network, voice  
10 data packets may be dropped in-transit to the destination due to inadequate or unavailable capacity of portions of data networks may result in gaps, silence, and clipping of audio at the receiving end.

A need thus exists for a method and system to improve the quality of voice calls or other audio communications over data networks.

**SUMMARY**

An aspect of managing the quality of telephony communications over a data network is call admission control according to the present invention. A call admission procedure consistent  
5 with the present invention determines whether to accept a call request from an originating terminal based upon the network condition. If a data network, or any portion of the data network, has become saturated with traffic (both audio and traditional data packet traffic), then according to the present invention, further call requests may be denied or rerouted to ensure some predetermined quality of service for the voice data packets.

10 According to one embodiment of the invention, call admission can only be granted if the throughput measurement response from a throughput measurement request matches or exceeds the throughput requirement of the call. An advantage of this embodiment's approach is that the network resources servicing the call are monitored while the call is set up or in progress to ensure an acceptable quality of service.

15 To establish a call between two or more terminals for performing telephony communications, according with this embodiment, a call request is sent from an originating terminal to the connection manager for processing. The call request may include the IP address of the originating terminal, an identifier of the destination terminal, a throughput requirement for the call, and optionally a list of one or more resource elements supported by the originating  
20 terminal to be used during an established call. Examples of resource elements include types of codecs (coders/decoders), the size of packets carrying audio data, and other resource elements.

Consistent with this embodiment, the call request defines a throughput requirement.

References to the throughput requirement may include bandwidth requirements, supported resource element requirements, size of packet requirements, quality of service requirements, and the like. The call request need not take place only on call initiation to set up the call; it may also  
5 take place during a call reallocation phase, to select alternative resource elements while the call is in progress.

In yet another embodiment of the present invention, an originating terminal communicates with the connection manager over the data network for call control signaling (to set up and terminate a call). After a connection is established between terminals over the data  
10 network, the terminals directly communicate media traffic (voice or other audio) and optionally, media traffic signaling with each other through the data network.

The connection manager performs call setup processing, which includes translation of dialed digits (such as 10-digit telephone number) to an IP address of a destination terminal. The connection manager also keeps tracks of the status (e.g., busy, idle, down, and so forth) of the  
15 terminals that it is responsible for. In addition, the connection manager keeps track of the usage of the transmission facility (the data network and other transmission resources) by the telephony application.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The invention will best be understood by reference to the following detailed description when read in conjunction with the accompanied drawings, wherein:

5           FIGURE 1 is a block diagram of a first embodiment of a telephony communications system in which voice or other audio data may be communicated over packet-based data networks;

10           FIGURE 2 is a block diagram of a second embodiment of a telephony communications system in which voice or other audio data may be communicated over packet-based data networks comprising communities (local area networks) connectively coupled via a wide area network;

15           FIGURE 3 illustrates the messages communicated between the various entities involved in call establishment according to the second embodiment of the invention;

            FIGURES 4A-4B illustrates the steps performed by the connection manager in the call establishment process;

            FIGURE 5 illustrates the messages communicated between the various entities involved in call establishment according to a third embodiment of the invention;

            FIGURE 6 illustrates components in a terminal and connection manager according to the first and second embodiments of the invention;

20           FIGURES 7-8 illustrate E-models that map conditions of a network link to a desired quality of service in accordance with a fourth embodiment;

FIGURE 9 illustrates a flow for processing a call request in accordance with the fourth embodiment that utilizes the E-models of FIGURES 9-10;

FIGURE 10 illustrates a telephony communications system that includes a plurality of communities and links between the communities over which the call admission control is

5 performed in accordance with a fifth embodiment; and

FIGURES 11A-11B illustrates the flow for managing a call request between terminals in different communities shown in FIGURE 10.

**DETAILED DESCRIPTION**

In the following description, numerous details are set forth to provide an understanding of the present invention. However, it will be understood by those skilled in the art that the present invention may be practiced without these details and that numerous variations or modifications from the described embodiments may be possible. For example, although the description refers to telephony communications over data networks, certain aspects of the methods and apparatus described may be advantageously used with other types of communications systems with different architecture.

Referring to FIGURES 1 and 2, a telephony communications system consistent with the present invention includes a number of endpoints or terminals (terminals 14, 16, 30 and 34 illustrated) that are capable of performing voice or other audio communications over a packet-based or message-based data network 20. As used here, "telephony communications" refers to the transmission and receipt of sounds (e.g., voice or other audio signals) between different points in a system using either wireline or wireless links. Example endpoints or terminals 14, 16, 30 and 34 may include computer systems with speech capability, telephone units that include interfaces to the data network 20, gateways coupled to standard telephones 34 through a public switched telephone network (PSTN) 32, and other types of communication devices. Telephony communications can occur between any two or more terminals over the data network 20.

The data network 20 may include, as examples, local area networks (LANs), metropolitan area networks (MANs), wide area networks (WANs), private networks such as Intranets and public networks such as the Internet, or a combination thereof. More generally, as

used here, a data network is any communications link that utilizes message-based or packet-based transmission of data or signaling information. Data network 20 may communicate according to the Internet Protocol (IP), as described in Internet Engineering Task Force (IETF) Request for Comment (RFC) 791, entitled "Internet Protocol," dated September 1981. Data  
5 network 20 may include a single network or link or multiple networks or links that are coupled through gateways, routers, edge routers and the like, as shown more clearly in FIGURE 2.

Shown in FIGURE 2 is local area network (LAN) network 21 communicatively coupled to LAN network 23 via edge router 80, wide area network 22 and edge router 40. LAN 21 may be considered as one community and LAN 23 may be considered as another community, as will be  
10 described in further detail herein.

A connection manager 50 is coupled to the data network 20 to manage telephony communications (e.g., call setup, processing, and termination) between or among the terminals 14, 16, and 30 (and other terminals shown or not shown in the figures). A policy server 60 may be queried by the connection manager 50 to determine the available bandwidth and other usage  
15 policy for telephony communications over the data network 20 to control the quality of service on the data network 20. Additionally, a network monitor system 19 may be coupled to the data network 20 to monitor certain characteristics and conditions of one or more portions of the data network 20. The characteristics and conditions monitored may include packet delay, jitter, and packet loss. Packet delay refers to a delay experienced in transmission due to processing,  
20 packetization, coding, decoding and queuing delays. Packet loss refers to the percentage loss of packets. Jitter refers to variations in the delay of a sequence of packets in the same stream of transmission. Jitter may contribute to end-to-end delay on a network because the receiving

platforms need to buffer the received data packets to compensate the different delays of packets. Furthermore, jitter may also affect the packet loss.

An edge router 40 is accessible through data network 20. Edge router 40 connects local area network 23 to the IP backbone or an external data network such as a WAN 22. Edge router 40 runs routing protocols and computes routing tables to be used for forwarding IP packets. In addition, edge router 40 can also forward packets, do tunneling, authentication, filtering, packet accounting, traffic shaping and address translation. Consistent with the present invention, there may be more than one edge router on the edge of each community network.

Although only one connection manager, policy server and edge router are illustrated in each community, multiple connection managers, edge routers and policy servers may in fact be coupled to and operating the data network without detracting from the spirit of the invention. In this arrangement, each of the multiple connection managers may be responsible for managing call requests from a predetermined group of terminals, and each policy server may be responsible for maintaining usage policy and available bandwidth for different portions of the data network

20. Further, more than one network monitor 19 may be included in the telephony communications systems of a community. For example, multiple network monitors may be located to enable monitoring of characteristics and conditions of different portions of the data network 20. A connection manager, policy server, edge router and network monitor may be implemented as interdependent threads executing on separate processing platforms or in a processing platform including some or all of the aforementioned components.

To establish a call between two or more terminals for performing telephony communications, in the embodiments of FIGURES 1 and 2, a call request is sent from an

originating terminal to the connection manager 50 for processing. The call request includes the IP address of the originating terminal P1, an identifier of the destination terminal, a throughput requirement for the call, and optionally a list of one or more resource elements supported by the originating terminal to be used during an established call. The identifier of the destination

5 terminal is a unique identifier associated with the terminal, such as a telephone number, an IP address P2, an e-mail address, a URL, or any other unique identification code associated with the terminal. A terminal may be relatively busy at the time a call is desired. As a result, processing capability and storage capability in the originating terminal may be limited so that resource elements that require high bandwidth are not indicated as being supported. Examples of resource  
10 elements include codecs (coders/decoders), the size of packets carrying audio data, and other resource elements.

In the event that the unique identifier carried by the call request does not contain an IP address for the destination terminal, the unique identifier may be sent to connection manager 50, or any other network resource capable of performing the task, to look up a destination IP address  
15 P2 for the destination terminal. In the event that the destination terminal is not connected to an IP network, but to a PSTN, then the destination terminal that the call request is routed to is the PSTN gateway terminal 30. PSTN gateway terminal has an IP address associated with it and the call is routed using that associated IP address. The call request contains sufficient information concerning the destination terminal (e.g. a telephone number) for the PSTN to establish a call  
20 from PSTN terminal gateway, over the PSTN to the destination terminal.

The call request may define a throughput requirement. References to the throughput requirement may include bandwidth requirements, supported resource element requirements, size

of packet requirements, quality of service requirements, and the like. With regard to the bandwidth requirements, throughput is the average number of user payload bits which traverse a telecommunication path over a period of time and is considered a measure of the effective data rate of a communications system. The call request need not take place only on call initiation to  
5 set up the call; it may also take place during a call reallocation phase, to select alternative resource elements while the call is in progress.

In the embodiments of FIGURES 1 and 2, an originating terminal 14 communicates with the connection manager 50 over the data network 20 for call control signaling (to set up and terminate a call). After a connection is established between terminals over the data network, the  
10 terminals communicate media traffic (e.g. voice, video or other audio), and optionally media traffic signaling, with each other through the data network 20. The connection manager 50 performs call setup processing, which includes translation of dialed digits (such as a 10-digit telephone number) to an IP address of a destination terminal. The connection manager 50 also keeps tracks of the status (e.g., busy, idle, down, and so forth) of the terminals that it is  
15 responsible for. In addition, the connection manager 50 keeps track of the usage of the transmission facility (the data network 20 and other transmission resources) by the telephony application. As used here, "telephony application" refers to one or more sessions of voice or other audio communications between or among the various terminals.

Optionally and in addition, by querying the policy server 60, the connection manager 50  
20 may determine the available bandwidth of the data network links over which the call will be established and discards any resource elements that may be unsupportable. Further, the connection manager 50 may also query the network monitor 19 to determine the current

characteristics and conditions of the network. Selection of resource elements may thus further be based on the current characteristics and conditions of the network (e.g., delays being experienced by packets and percentage of packet loss).

Next, consistent with these embodiments, the connection manager 50 ranks the remaining supportable resource elements based on predetermined merit attributes, which may include quality of service, the available bandwidth, expected usage of transmission resources, and other attributes. Selection of the resource elements to use for a particular call may in addition be based on the ranking performed by the connection manager 50.

An aspect of managing telephony communications over a data network in accordance with the present invention is call admission control. A call admission procedure determines whether to accept a call request from an originating terminal. If a data network, or any portion of the data network, has become saturated with traffic (both audio and traditional data packet traffic), then further call requests may be denied to ensure some predetermined quality of service is maintained for the existing active calls. Call admission may be granted if the throughput measurement response from a throughput measurement request matches or exceeds the throughput requirement of the call. This may rely upon a test signal propagating between network resources. Alternatively, or in addition, call admission may be based on usage of links between different groups of terminals (with the groups referred to as communities). Each community may include multiple terminals that are capable of communicating with each other without being subjected to call admission control. This is made possible by grouping terminals that are coupled to high capacity links, such as LANs. Thus, within each community, voice calls between terminals are allowed to proceed when requested. To provide some limitation on

bandwidth usage of the communication link within each community, resource element selection (such as the codec and packet size selection described above) may be used to limit the bandwidth of each call session when large numbers of call sessions are present in the community. The call admission control may be provided for calls by using a throughput measurement response that satisfies the throughput measurement requirement and/or based on the monitored usage of the links among the communities.

One type of resource element is the audio coder/decoder (codec), which may be used by each of the terminals involved in a call session. As shown in Figure 6, an audio codec encodes audio signals originating from an audio input device (e.g., microphone) for transmission and decodes received audio data for output to an output device (e.g., a speaker). The codec may be implemented in software. Several types of codecs are available that have varying levels of data compression and data transfer rate requirements. For example, the G.711 codec provides uncompressed communications of voice data, but has a data transfer rate requirement of 64 kbps (kilobits per second). Other codecs, such as those conforming to the G.728, G.729A, G.729, G.723.1, and G.722 recommendations have varying compression algorithms and data transfer rate requirements, which are typically lower than that of the G.711 codec. The listed G series of audio codecs are recommendations from the International Telecommunications Union (ITU).

Generally, higher compression leads to a reduced amount of data so that data transfer rate requirements over a link may be reduced. However, because compression of data may cause loss of original voice fidelity, audio quality may be adversely affected. Thus, the two objectives of higher quality audio and lower data transfer rate requirements may conflict.

Conventionally, an originating terminal that desires to establish a voice communication with a destination terminal sends a list of supported codecs to the destination terminal. In response, the destination terminal chooses an acceptable codec from the list. Such a process is provided by the ITU H.323 protocol, which is a recommendation for packet-based multimedia.

5 Although such a process allows voice communications employing a commonly supported codec between the originating and destination terminals, it does not take into account the capacity and usage of the link and other transmission resources between the terminals, in this case the data network 20, as well as other transmission or communications resources.

10 Another resource element is the packet size supported by the codec to communicate voice or other audio. A voice packet size refers to the duration of a speech sample that is contained in each packet. For example, a packet size may be 10 milliseconds (ms), which indicates that a 10-ms sample of speech is contained in each packet. Other packet sizes include 20 ms, 40 ms, and so forth. Selection of the packet size has implications on the burden placed on the data network in a given call session. Shorter packets generally are associated with higher overhead, since more  
15 audio data packets are communicated over the data network 20 between terminals. Longer packets are associated with reduced overhead, but come at the cost of longer delays at the originating station since a longer speech sample is created between successive transmissions of audio over the data network 20. Thus, selection of packet size may also lead to a conflict  
20 between the objectives of higher quality audio and reduced load on the data network 20 and other transmission resources.

A call admission control mechanism implemented in the terminals, connection manager(s), policy server(s), and/or network monitor(s) of the telephony communications system

10 balances the need for high audio quality as well as the need to reduce burden on the data  
network 20 and other transmission resources. The call control mechanism, by analyzing  
throughput of the communication path(s), may select a supported codec, packet size, and/or other  
resource element that takes into account support for the resource element by communicating  
5 terminals, the available capacity of the data network 20 and other transmission resources, and the  
objective to achieve the highest possible quality of service.

The present invention may be configured to comply with quality of service (QoS)  
protocols such as RSVP, Diffserv and also MPLS. It should be noted that these QoS protocols  
may be used in combination, for example, Diffserv may be used in combination with MPLS.  
10 FIGURE 5 illustrates the flow for processing a call request between an originating terminal and a  
destination terminal in accordance with an RSVP enabled embodiment.

Different protocols exist that define packet-based call control signaling for call sessions  
over packet-based networks. One example call control protocol is a Session Initiation Protocol  
(SIP), which is used to initiate call sessions as well as to invite members to a session that may  
15 have been advertised by some other mechanism, such as electronic mail, news groups, web  
pages, and other mechanisms. SIP is part of the multimedia data and control architecture from  
the IETF. A version of SIP is described in RFC 2543, entitled "SIP: Session Initiation Protocol,"  
dated 1999. The other protocols in the IETF multimedia and control architecture include the  
Resource Reservation Protocol (RSVP), as described briefly below and also in RFC 2205, for  
20 reserving network resources; the Real-Time Transport Protocol (RTP), as described in RFC  
1889, for transporting real-time data and providing quality of service (QoS) feedback; the Real-  
Time Streaming Protocol (RTSP), as described in RFC 2326, for controlling delivery of

streaming media; the Session Description Protocol (SDP), as described in RFC 2327, for describing multimedia sessions; and the Session Announcement Protocol (SAP) for advertising multimedia sessions by multicast. Another signaling protocol that may be used includes MGCP as referred to in IETF document RFC 2705, entitled "Media Gateway Control Protocol" dated  
5 October 1999. MGCP is evolving to a new signaling protocol, which may also be used, called "Megaco", as referred to in ITU H.248.

The Resource Reservation Protocol (RSVP) is an Internet Engineering Task Force (IETF) standard designed to support resource (for example, bandwidth) reservations through networks of varying topologies and media. Through RSVP, a user's quality of service requests are  
10 propagated to all routers along the data path, allowing the network to reconfigure itself (at all network levels) to meet the desired level of service. The RSVP protocol reserves network resources by establishing flows throughout the network. A flow is a network path associated with one or more senders, one or more receivers and a certain quality of service. A sending host wishing to send data that requires a certain Quality of Service (QoS) will send, via an RSVP-  
15 enabled Service Provider, "path" messages toward the intended recipients. These path messages, which describe the bandwidth requirements and relevant traffic parameters of the data to be sent, are propagated to all intermediate routers along the path. A receiving host, interested in this particular data, will confirm the flow (and the network path) by sending "reserve" (RESV) messages through the network, describing the bandwidth characteristics of data it wishes to  
20 receive from the sender. As these reserve messages propagate back toward the sender, intermediate routers, based on bandwidth capacity, decide whether or not to accept the proposed reservation and commit resources. If an affirmative decision is made, the resources are

committed and reserve messages are propagated backward on the previously traversed path from source to destination.

Diffserv relies on traffic conditioners sitting at the edge of the network to indicate each packet's requirements, and capability can be added via incremental firmware or software upgrades. An embodiment utilizing Diffserv, for example, takes the IP TOS (type of service) field, renames it the DS byte, and uses it to carry information about IP packet service requirements. Since it operates on Layer 3 of the OSI network model, it makes no assumptions about the underlying transport. Diffserv is suitable for serving a community, such as an enterprise network (e.g. LAN 23) at the point where customer traffic meets a service provider network (e.g. WAN 22), and because it specifies QOS at Layer 3 it can run without modification on any Layer 2 infrastructure that supports IP. Thus, Diffserv may be used in conjunction with MPLS. Further, traffic-engineered Diffserv may be used in conjunction with RSVP. Thus, routers in the WAN need not be concerned about bandwidth reservation for each call, which will be explained further later in this description.

Multiprotocol Label Switching (MPLS) is an evolving IETF standard intended for Internet application. MPLS is a method of speeding up IP-based data communication networks by routing at the edge and switching at the core. In other words, routers are used at the ingress and egress edges of the network, where their high levels of intelligence can be best used and where their inherent slowness can be tolerated. Switches are used in the core of the network, where they can take advantage of the intelligent switching labels provided by the routers, and where their inherent speed offers great advantage. MPLS enablement may be implemented as follows: As an IP data stream enters the edge of the network, the ingress router reads the full

address of the first data packets and attaches a small “label” in the packet header, which precedes the packet. The switches in the core of the network examine the much-abbreviated label, and switch the packet with much greater speed than if they were forced to consult programmed routing tables associate with the full IP address. All subsequent packets in a data stream are automatically labeled in this fashion; and very quickly, as they have been anticipated. MPLS requires a network of sophisticated label-switching routers capable of reading header information and assigning packets to specific paths like virtual circuits on a switched network. MPLS specifies ways that Layer 3 traffic can be mapped to connection-oriented Layer 2 transports like ATM and frame relay; it adds a label containing specific routing information to each IP packet and allows routers to assign explicit paths to various classes of traffic. It also offers capabilities for traffic-engineering which can boost IP traffic efficiency by re-routing congested traffic over under-utilized links to alleviate network congestions. The ability to perform traffic-engineering using MPLS has the potential to improve the quality of network services when the traffic volume is very high.

Referring to FIGURES 3 and 4A-4B, alternative processes for establishing a call between an originating terminal (e.g., the terminal 14, also referred to as terminal P1) and a destination terminal (e.g., the terminal 16, also referred to as terminal P2) shall now be discussed.

FIGURE 3 illustrates the messages communicated between the various entities involved in call establishment according to a first embodiment, and FIGURES 4A-4B further illustrates the tasks performed by the connection manager 50 in the call establishment process according to the first embodiment.

To start a call, the originating terminal 14, which has an IP address P1, sends a call request, such as a CALL\_SETUP message, which is received (at step 102 in FIGURE 4A) by the connection manager 50. The CALL\_SETUP message includes a number identifying the destination terminal, and optionally, a codec list including the codecs that are supported by the terminal 14, a list of supported packet sizes, and/or a list of other supported resource elements. Example codecs that are supported include G.711, G.728, G.729, G.729A, G.723.1, and G.722 codecs. The G.711 codec communicates uncompressed audio data and requires a 64-kbps data transfer rate, whereas the other codecs provide varying levels of data compression with lower data transfer rate requirements. For example, the G.728 codec requires a 16-kbps transfer rate, the G.729 codec requires an 8-kbps transfer rate, and the G.723.1 codec requires a transfer rate of 6.3 kbps, 5.3 kbps, or less.

In the connection manager 50, the list of available codecs and list of supported packet sizes in the CALL\_SETUP message are received (at 114) and combined into a candidate list. Alternatively, the different lists of resource elements may be maintained as separate candidate lists. For example, the CALL\_SETUP message sent to connection manager 50 may also send a list of available codecs and packet sizes from terminal 14 as a candidate list.

Based on the CALL\_SETUP message, the connection manager 50 translates (at 104) the number (e.g., the dialed number) of the called party into an IP address (e.g., address P2 of the destination terminal 16). This is returned to originating terminal 14 as "REPLY\_P2" as shown in FIGURES 3, and it may include a throughput measurement request.

A throughput measurement request may comprise a trace route and/or a trace packet, or alternatively a request for a trace route and/or trace packet to be sent from a particular device. A

Trace route may be used to build a list of the hops and network resources traversed from an originating network resource on a link to another network resource on data network 20. A list may be generated comprising the hops and the network resources that the trace route goes via.

The listed network resources and the respective hops between resources may then be monitored

5 for congestion. A trace route may propagate from terminal 14 with IP address P1 and terminal 16 with IP address P2 (at 106) to discover the full connection path between a network resource and another. TRACE\_ROUTE\_REQUEST may be sent from terminal 14 and may propagate via

network resources on a connection path between terminal 14 and terminal 16. A

TRACE\_ROUTE\_RESPONSE then propagates back from terminal 16 to terminal 14. It should

10 be noted that TRACE\_ROUTE\_REQUEST may follow a different connection path to

TRACE\_ROUTE\_RESPONSE.

A throughput measurement response such as a RESOURCE\_INFO message may be sent from a network resource (e.g. originating terminal 14) to connection manager 50. Alternatively or additionally, a throughput measurement response may be provided from information obtained

15 from the aforesaid trace packet, which may be used to provide RESOURCE\_INFO messages directly or indirectly from each network resource on a connection path. The RESOURCE\_INFO message may comprise the aforementioned list comprising the hops and/or the network resources that the trace route goes via. The list may change dynamically to include a different combination of network resources if it is necessary to reroute the call via other network resources for any

20 reason.

Connection manager 50 then tests if the throughput measurement response meets a throughput measurement requirement. It may do this by monitoring the network resources in the

resource list from the trace route. Thus, the connection manager 50 may maintain the QoS for the call by monitoring the network resources in the list for congestion and/or monitoring that the throughput measurement response remains to meet a throughput measurement requirement over the life of the call. A throughput measurement requirement may comprise performance

5 thresholds such as time for a test packet to propagate from one network resource to another, available bandwidth, congestion, efficiency of communications link and other performance metrics. These performance metrics may be related to a service level agreement, as referred to later. If the throughput measurement response substantially meets this threshold, then the next phase of call setup may proceed.

10 If the throughput measurement response does not substantially meet the threshold then the connection request over the data network may be rejected (at 110). If this happens, an alternative resource may be used for the communications link, such as a switched network (e.g. PSTN), another data network, a connection-oriented network such as ATM or Frame Relay, a virtualized circuit or a dedicated private link. If a PSTN is used, the data packets may be

15 packaged to travel over the PSTN and unpackaged to travel over the data network of the destination community. In such connection-oriented networks, packets are received in the same order in which they were transmitted.

Optional steps of policy service processing (as indicated by broken lines) may also be enacted, as a contingent method of providing QoS if a trace-route enabled QoS path is unable to

20 be established, as represented on FIGURE 3 by broken lines. Optionally, the connection manager 50 may send (at 116) a QUERY\_BANDWIDTH\_POLICY message to the policy server 60. The QUERY\_BANDWIDTH\_POLICY message may include the IP addresses of the originating and

destination terminals (P1, P2) and a request for the real time congestion status and linkage usage of appropriate edge routers which may serve originating and destination terminals (P1, P2).

Optionally, also included in the message is a query for allowable bandwidth and other usage policy for a telephony application between the pair of terminals at the present time. The policy

server 60 may also set the percentage of use for telephony communications and the percentage for traditional packet data communications.

Policy server 60 may contain call admission policy for connections between predetermined originating and destination terminals and/or between particular network resources. Such policy may be set by Service Level Agreements (SLAs). These are agreements between a user or a community and a service provider, defining the nature of the service provided and establishing a set of metrics to be used to measure the level of service provided measured against the agreed level of service. Typically, a SLA may be set up over a WAN to provide a predetermined quality of service between communities. Such service levels might include provisioning (e.g. quality of service), average availability, restoration times for outages, availability, average and maximum periods of outage, average and maximum response times, latency, delivery rates (e.g. average and minimum throughput). An SLA policy may, for example, agree to deliver 1% packet loss, with 5ms average delay and 50ms packet size.

The policy server 60 responds to the query by sending a `REPLY_BANDWIDTH_POLICY` message back to the connection manager 50 to indicate the real time congestion status and linkage usage of network resources which may serve terminals P1 and P2, and also optionally, the available bandwidth that may be allocated between the terminals P1 and P2 for the present call session. Based on the received allocated bandwidth, the connection

manager 50 updates (at 118) the candidate list by deleting unacceptable codecs. The policy server 60 may allocate a reduced bandwidth for the telephony application because of high traffic carrying traditional data packets (e.g., e-mail traffic, web browsing traffic, file transfer traffic, and so forth). Thus, codecs that have high bandwidth requirements may be deleted (at 118) from the candidate list. Examples of such high bandwidth codecs include the G.711 codec. In addition, unacceptable packet sizes are also deleted by the connection manager 50 from the candidate list (at 118), depending on the available bandwidth. If a limited bandwidth is available, then shorter packet sizes may be deleted from the list by the connection manager 50. Thus, the connection manager 50 selects one or more resource elements (e.g., codecs and packet sizes) that are supported based on the available bandwidth of the data network 20.

Further alternatively or in addition, the connection manager 50 can also perform an additional bandwidth restriction based on the usage of transmission resources. Consistent with the present invention, each connection between a pair of terminals shares a pool of transmission resources (links coupling the terminals that the connection manager 50 is responsible for, edge routers, routers and gateways coupling the links, and other resources) with other applications. The connection manager 50 keeps track of the usage of the pool of transmission resources by tracking the number of voice calls and other traffic. When the usage reaches a predetermined threshold, the connection manager 50 may further limit the bandwidth usage. The connection manager 50 may use this limitation to further delete (at 120) unacceptable codecs, packet sizes, and other resource elements from the candidate list so that a further reduced number of resource elements may be selected.

The connection manager 50 may then send (at 122) a query message, e.g., a QUERY\_RESOURCE\_AVAILABILITY message, to the destination terminal P2 to identify the supported codecs, packet sizes, and other resource elements in the destination terminal P2.

The results are returned in a REPLY\_RESOURCE\_AVAILABILITY message, from which the connection manager 50 can determine the codecs, packet sizes, and other resource elements supported by the destination terminal P2. The candidate list of codecs, packet sizes, and other resource elements is updated within the connection manager 50 based on the available codecs in the destination terminal P2, with unsupported codecs, packet sizes, and other resource elements deleted from the candidate list.

Potentially, all codecs or packet sizes may have been deleted from the candidate list. If either the list of codecs or the list of packet sizes is empty (as determined at 124), then no supported codec or packet size exists to allow a call to proceed between the terminals P1 and P2, at which point the connection manager 50 sends (at 126) a message to terminate the call setup. The connection manager 50 may also inform the originating terminal P1 of the setup failure.

If at least one codec and at least one packet size is available in the candidate list, then the call may proceed. If two or more codecs are present in the candidate list, then the codecs are reordered (at 128) by applying a merit-based codec ranking algorithm to rank the codecs in the candidate list in the descending merit order (described further below). Packet sizes may also be ordered according to a merit ranking algorithm, as may other resource elements.

The codec, packet size, and other resource element having the highest relative rank is selected. Alternatively, selection may be performed by the terminals, which may be adapted to select the highest ranking resource elements from a list.

Next, assuming that resources are available, the connection manager 50 sends a CALL\_SETUP message to the destination terminal P2 (at 130), with the Call Setup message including an identifier of the calling party (either the calling terminal's telephone number or its IP address), the selected codec, packet size, and other resource element. The connection manager 50 then proceeds (at 132) to the remaining tasks to be performed in the call setup, including sending a CALL\_ADMISSION\_RESPONSE message identifying the selected codec, packet size, and other resource element back to the originating terminal P1. Alternatively, the codec, packet size, and other resource element may be communicated as parameters in a Setup Connection message sent by the connection manager 50 to connect the call between terminals P1 and P2. A media path is then set up between the terminals P1 and P2.

Although reference is made to selection of several resource elements, it is contemplated that further embodiments may select fewer than all the possible types of resource elements in the call management process. For example, connection manager 50 may perform selection of only codecs to manage bandwidth usage and quality of service on the data network 20. In addition, if multiple connection managers are present in the data network 20, then communications may occur between connection managers 50 to enable selection of resource elements for establishing a call between terminals controlled by the connection managers.

FIGURE 5 illustrates the messages communicated between the various entities involved in call establishment using Resource Reservation Protocol (RSVP) signaling. As explained above, RSVP is an IETF standard designed to support resource (for example, bandwidth) reservations through networks of varying topologies and media. Through RSVP, a user's quality of service requests are propagated to all routers along the data path, allowing the network to

reconfigure itself (at all network levels) to meet the desired level of service. More information on RSVP may be found in RFC 2205, which is available at the IETF repository located at [www.ietf.org](http://www.ietf.org).

To start a call, the originating terminal 14, which has an IP address P1, sends a call request, such as a CALL\_SETUP message, which is received by the connection manager 50. The CALL\_SETUP message includes a number identifying the destination terminal, and optionally, a codec list including the codecs that are supported by the terminal 14, a list of supported packet sizes, and/or a list of other supported resource elements.

In the connection manager 50, the list of available codecs and list of supported packet sizes in the CALL\_SETUP message are received and may be combined into a candidate list. Alternatively, the different lists of resource elements may be maintained as separate candidate lists. A reader skilled in the art to which this invention pertains will appreciate that these steps may be performed in an alternative order to those described.

Based on the Call Setup message, the connection manager 50 translates the number (e.g., the dialed number) of the called party into an IP address (e.g., address P2 of the destination terminal 16). It may also send the recommended codec type and packet size which can be used for bandwidth information in addition to the IP address which is used by the RSVP PATH message.

Next, in accordance with FIGURE 5, an RSVP 'PATH' signal may be sent from originating terminal 14 to network resources, such as but not limited to, edge router 40, edge router 80 and destination terminal 16. RSVP signaling to the various network resources may happen simultaneously or in a sequence. The RSVP protocol engages network resources by

establishing flows throughout the network. A flow is a network path associated with one or more senders, one or more receivers and a certain quality of service. A sending host wishing to send data that requires a certain Quality of Service (QoS) will send, via an RSVP-enabled Service Provider, "path" messages toward the intended recipients. These path messages, which describe the bandwidth requirements and relevant parameters of the data to be sent, are propagated to all intermediate routers along the path. A receiving host, interested in this particular data, will confirm the flow (and the network path) by sending reserve (RESV) messages through the network, describing the bandwidth characteristics of data it wishes to receive from the sender. As these RESV messages propagate back toward the sender, intermediate routers, based on bandwidth capacity, decide whether or not to accept the proposed reservation and commit resources. If an affirmative decision is made, the resources are committed and RESV messages are propagated to the next hop on the path from source to destination.

In situations that an ATM, frame relay, or an engineered DiffServ core (WAN) network for communications between communities, it is preferred not to use RSVP in the core. In these scenarios, the RSVP messages still traverse the network end to end, but routers in the core will not examine RSVP messages and treat them like any other data packets. This process keeps RSVP benefits on calling and called party communities, and leaves the core network free of RSVP states overhead.

Since RSVP is unidirectional, terminal 16 may also establish RSVP-enabled communications by sending RSVP path messages in the opposite direction to the RSVP arrows shown. Although this has not been shown on FIGURE 5, an RSVP-enabled communication path

from terminal 16 to terminal 14 may be established in a similar fashion to that described for that set up from terminal 14 to terminal 16.

Next, an RSVP status message is sent from terminal 14 to connection manager 50 confirming that resources on the communications path have been reserved. Connection manager 50 then sends a call admission response to terminal 14 and terminal 16 to admit the call. A media path may then carry the media traffic.

Referring to FIGURE 6, components of an example terminal and connection manager are illustrated. In FIGURE 6, the components of the terminal 14 and connection manager 50 described above are illustrated in more detail. As noted above, the terminal 14 can be one of many types of devices capable of communicating voice over the data network 20. These terminals may include computer systems, telephones that are configured to communicate over a data network, a gateway system to the public switched telephone network (PSTN), and other communications devices. The layers of the terminal 14 include a network interface controller 302 that is coupled to the data network 20. Above the network interface controller 302 is a network device driver 304 and a network stack 306, such as a TCP/IP or UDP/IP stack. TCP stands for Transmission Control Protocol, and is described in IETF RFC 793, entitled "Transmission Control Protocol," dated September 1981. UDP stands for User Datagram Protocol, and is described in RFC 768, entitled "User Datagram Protocol," dated August 1980. Above the network stack 306 is a real time protocol (RTP) layer 308 that performs various tasks associated with real time communications such as telephony communications. Incoming data from the data network 20 is received through the layers 302, 304, 306 and 308 and routed to an audio codec 310, which has been selected from a number of available codecs as discussed above. The

incoming data is decoded by the codec 310 and routed to a digital-to-analog (D/A) converter 312 to produce the output at a speaker 314. Outbound data to the network 20 originates at a microphone 316 or from an application routine 318. A user can speak into the microphone 316 to communicate voice data over the data network 20. Alternatively, the application routine 318 (or  
5 some other routine) may generate voice, video or other audio data to be transmitted to the data network 20. Examples of this may include an automated answering application, such as a voice mail application or a voice prompt application from which users can select to access to various services.

From the microphone, audio signals are passed through an analog-to-digital (A/D)  
10 converter 320, which digitizes the audio signals and passes the digital audio data to the codec 310. The codec 310 encodes the data and transmits the coded data down layers 308, 306, 304, and 302 to the data network 20. The audio traffic is communicated through the data network 20 to another terminal to which the terminal 14, 16, or 30 has established a call connection.

In addition to the audio traffic path, a control path exists between the terminal and the  
15 connection manager 50 to set up, maintain, and terminate voice calls over the data network 20. In the terminal 14 one or more application routines 318 may generate control messages that are transmitted to the connection manager 50 through the network stack 306, network device driver 304, network interface controller 302, and the data network 20. Control signaling from the connection manager 50 is similarly received through the same layers from the data network 20  
20 back to the one or more application routines 318.

As shown in FIGURE 6, in the connection manager 50, similar layers may exist. A network interface controller 330 in the connection manager 50 is coupled to the data network 20.

Above the network interface controller 330 is a network device driver 332 and a network stack 334, such as a TCP/IP or UDP/IP stack. One or more call processing routines 336 in the connection manager 50 controls the management of calls between terminals that are assigned to the connection manager 50. The call processing routines 336 perform the establishment of calls, maintenance of calls, and termination of calls. The call processing routines 336 may also periodically determine the available bandwidth over the data network 20, which may cause it to update the codec and packet size used by the terminals in the voice communication session over the data network 20.

In each terminal and connection manager, various software routines or modules may exist, such as the one or more application routines 318, network stack 306, and device driver 304 in the (D/A) converter 312 to produce the output at a speaker 314. Outbound data to the network 20 originates at a microphone 316 or from an application routine 318. A user can speak into the microphone 316 to communicate voice data over the data network 20. Alternatively, the application routine 318 (or some other routine) may generate voice or other audio data to be transmitted to the data network 20. Examples of this may include an automated answering application, such as a voice mail application or a voice prompt application from which users can select to access to various services. From the microphone, audio signals are passed through an analog-to-digital (A/D) converter 320, which digitizes the audio signals and passes the digital audio data to the codec 310. The codec 310 encodes the data and transmits the coded data down layers 308, 306, 304, and 302 to the data network 20. The audio traffic is communicated through the data network 20 to another terminal to which the terminal 14 has established a call connection.

Various software routines or modules may exist, as shown in FIGURE 6. Instructions of such software routines or modules, and others, may be stored in storage units 344 and 349. in the terminal and connection manager , respectively. The storage units 344 and 349 may include machine-readable storage media including memory devices such as dynamic or static random access memories, erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs), and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; and optical media such as compact discs (CDs) or digital versatile discs (DVDs).

The instructions may be loaded and executed by control units 340 and 348 in the terminal and connection manager , respectively, to perform programmed acts. The control units 340 and 348 may include microprocessors, microcontrollers, application-specific integrated circuits (ASICs), programmable gate arrays (PGAs), or other control devices as is known in the art. The terminal 14 may also include a digital signal processor 346 for performing arithmetic intensive operations such as compression and decompression operations performed by the audio codec 310 as is known in the art.

The following discusses merit-based codec ranking. A modified ranking system may be provided for packet size and/or other resource element selection. The connection manager 50 maintains a table of characteristics of each codec including the following attributes: voice quality (Q), bandwidth usage (B), and terminal DSP (e.g., digital signal processor 346 in FIGURE 9) resource usage (R). The Q, B, and R attributes may contain numeric values (ranging between 0 and 1). The attribute B in one embodiment may represent the inverse of the actual bandwidth usage, that is, a higher B value indicates low bandwidth usage and a low B value indicates high

bandwidth usage. The attribute R similarly represents the inverse of the actual DSP usage. A merit factor M can be computed for each codec in the candidate list as a linear combination of the attributes Q, B, and R according to the following equation:

$$M=W_Q*Q+W_B*B+W_R*R,$$

5 where  $W_Q$ ,  $W_B$ , and  $W_R$  are weights that are assigned to the attributes Q, B, and R, respectively. The values of the weights  $W_Q$ ,  $W_B$ , and  $W_R$  may be dynamic and can be based on usage of the pool of transmission resources used for the telephony application. Thus, in one example, the values of the weights  $W_Q$ ,  $W_B$ , and  $W_R$  may be assigned as follows:

10 
$$W_Q=(1-t)*0.8, W_B=t, \text{ and } W_R=(1-t)*0.2,$$

where t is the percentage usage of the pooled transmission resources for the telephony application. The codecs in the candidate list may be arranged in descending order of the merit factor M, from which a codec can be selected for use in the call to be established.

15 Thus, the merit factor M may be higher for codecs having relatively high audio quality (Q), low expected bandwidth (e.g., data transfer rate) usage (B), and low expected DSP usage (R). Codecs having relatively low audio quality, high expected bandwidth usage, and high DSP usage will have a lower M value. Thus, generally, the value of the merit factor M is increased with higher audio quality and decreased usage of transmission resources (e.g., links in the data network 20 and DSP 346).

20 As noted above, in connection with FIGURES 1 and 2, the telephony communication system 10 includes a network monitor 19 for monitoring various characteristics and conditions of one or more portions of the data network 20. Multiple network monitors may be present for

monitoring different portions of the data network 20. The characteristics and conditions monitored include packet delay, jitter, and percentage of packet loss.

The network monitor 19 may perform monitoring of a network link in a number of different ways. One technique is to use a network monitor having two different nodes on a network link. One node of the network monitor can send test packets targeted to the other node in the network monitor 19. From the transmission and receipt (or lack of receipt) of test packets, the nodes of the network monitor 19 can determine the delays in transmissions of packets as well as the percentage of packet loss. The network monitor 19 can periodically communicate test packets to monitor the link on a periodic basis. Such a technique may be referred to as a static monitoring technique.

A dynamic technique to monitor a link is to access routers or gateways that communicate with the link. Routers and gateways maintain management information that keep track of delays being experienced with links that the routers and gateways are coupled to as well as amounts of packets that are being lost. Thus, each time a connection manager accesses a network monitor to request the current characteristics and conditions of a particular link, the network monitor can issue a query to a particular gateway or router to determine the current conditions.

Once the packet delay and loss information is determined by the connection manager 50, the connection manager 50 can access a database of models (referred to as E-models) for each connection manager to determine if a codec can satisfy a desired level of quality based on the prevailing network link conditions. E-models may also be maintained for the other resource elements. Two E-models 350 and 352 are illustrated in FIGURES 7 and 8 for the G.729A and G.723.1 codecs, respectively. Each E-model includes a chart mapping packet delays and

percentage of packet loss to a desired quality level. In each E-model chart 350 or 352, an R-value represents the desired quality of service. The connection manager 50 may maintain profiles and policies establishing the desired R-values of calls between different combinations of callers. For example, for internal calls within an organization, a lower quality of service (and therefore lower R value) may be established, whereas external calls are set at higher R values. Other embodiments may use different representations of the quality of audio service of codecs and other resource elements.

In the chart 350 for the G.729A codec, the horizontal axis represents packet delay and the vertical axis represents the R value. The curves 360A-360I represent different percentages of packet losses. In one example, the curve 360A represents a 0% packet loss, the curve 360B represents a 0.5% packet loss, the curve 360C represents a 1% packet loss, the curve 360D represents a 1.5% packet loss, the curve 360E represents a 2% packet loss, the curve 360F represents a 3% packet loss, the curve 360G represents a 4% packet loss, the curve 360H represents an 8% packet loss, and the curve 360I represents a 16% packet loss. Thus, as illustrated in FIGURE 7, the higher the delay and percentage packet loss, the lower the R value. An R value of 90 generally indicates that users are very satisfied, an R value of 80 generally indicates that users are satisfied, an R value of 70 generally indicates that some users are dissatisfied, an R value of 60 generally indicates that many users are dissatisfied, and an R value of 50 and below generally indicate that nearly all users are dissatisfied with the level of service. An R value may also be used as a metric in a Service Level Agreement, as discussed earlier.

The chart 352 in FIGURE 8 for the G.723.1 codec is similar to the chart 350 in FIGURE 7, with the curves 362A-362I representing corresponding percentages of packet loss to curves 360A-360I in FIGURE 7. Thus, given the current packet delay and percentage of packet loss, the E-models for the various codecs may be accessed to determine which codec can support the  
5 desired R value. In further embodiments, different models may be used for codec or other resource element selection.

Thus, referring to FIGURE 9, in accordance with an alternative embodiment that uses E-models, such as 350 and 352 as described previously, the connection manager 50 receives (at  
10 370) a call request from an originating terminal. The call request may identify the resource elements, including codecs, supported by the originating terminal. The connection manager can perform (at 371) selection of the codecs and other resource elements based on the available bandwidth and usage of transmission resources, including the data network 20, as described above in connection with FIGURES 3-5. This may reduce the number of codecs and other resource elements.

15 Further, based on the profiles and policies associated with the identified originating and destination terminals in the call request, the connection manager identifies (at 372) the target quality of service (R value). Next, a TRACE\_ROUTE signal (at 373) may be sent from the originating terminal to at least one endpoint, and then receive a list of network resources that the signal traverses from end to end. The connection manager 50 can send (at 374) query messages  
20 to the network monitor 19 to determine the current characteristics and conditions of the network resources in the list (from the trace route), including delay and packet loss. Based on the identified delay and packet loss information, the connection manager 50 accesses (at 376) the E-

models of the supported codecs. From the E-models, the connection manager 50 identifies (at 378) the codecs and other resource elements that satisfy the target R value. Next, the codecs and other resources may be ranked (at 380) as described above based on various merit attributes to enable selection of one of the codecs and other resource elements to use during the call, as

5 described above.

Some embodiments of the invention may provide one or more of the following advantages. A flexible codec (and other resource element) selection strategy is provided to enforce a policy based on the codec data rate between a pair of terminals where the codec (and other resource element) selection takes into account the capacity and resource limitation of the

10 terminals as well as network traffic load and actual transmission resource usage in each terminal. Selection of resource elements may also be based on the prevailing characteristics and conditions of the network, such as delay and packet loss. Fine policy control over telephony traffic over a data network is made available. Selection may be biased towards high voice quality when traffic is light; however, if other network traffic high, then voice quality may be reduced to reduce the

15 load on the data network.

The codec and other resource element selection technique and apparatus may be used with other applications. For example, for video conferencing communications sessions over a packet-based data network, selection of video codecs may also be used to reduce load on the data network.

20 Referring to FIGURE 10, one arrangement of a voice communication system 400 that includes communities is illustrated. In FIGURE 10, each of the three communities 402, 404, and 406 includes its set of terminals. In the community 402, terminals 408 are coupled to a link 409 (e.g., a WAN, or other network). A connection manager 410 is also coupled to the link 409 to

manage calls between or among the terminals 408 and between one or more of the terminals 408 and a terminal external to the community 402. An edge router 412 couples the internal link 409 to an external link 432, which couples the first community 402 to the second community 404. In the second community 404, terminals 414 are coupled to an internal link 415. An edge router 418 is connected between the internal link 415 and the external link 432. The edge router 418 also couples the internal link 415 of the second community 404 to another external link 430. The external link 430 is coupled to a third community 406. Inside the community 406 is an edge router 422 that couples the link 430 to an internal link 421, which is connected to terminals 420. The second and third communities 404 and 406 share a connection manager 416, which manages calls within each of, or between, the communities 404 and 406 as well as between a terminal in one of the communities 404 and 406 and another community, such as community 402.

Generally, the links 430 and 432 (and other external links connecting communities) have lower bandwidths than the internal links in each of the communities. However, it is contemplated that exceptions to this exist where an external link may have higher bandwidth than an internal link. For a given community  $I$ , the following parameters may be defined:  $L_I$ , which represents the limit on a total bandwidth between the community and a device or system external to the community;  $M_I$ , which represents the threshold at which reselection of a codec, packet size, or other resource element is performed to reduce load on a link in a community;  $N_I$ , which represents a threshold to restrict outgoing calls, or a throughput requirement; and  $T_I$ , which represents the usage of the transmission resources in the community, or a throughput measurement.

Thus, outgoing new call requests from the community I may be denied if the value of  $T_I$  exceeds the threshold  $N_I$ . If the traffic  $T_I$  exceeds the threshold  $M_I$ , then the connection manager for the community I can start to perform codec and other resource selection to reduce traffic. As described above, a connection manager may discard codecs and/or other resource elements based on transmission resources that the connection manager monitors, including the several thresholds  $L_I$ ,  $M_I$ , and  $N_I$  of the community I. By example, the value of  $M_I$  may be about 60% to 80% of  $L_I$ . The value of  $N_I$  can be set at a value closer to or at  $L_I$ . In some embodiments, if the throughput measurement request does not meet the throughput measurement requirement, then the call may not be admitted. Optionally, if the call cannot be admitted then connection manager may iteratively discard codecs and/or other resource elements until the throughput requirement of the call request is met by the throughput measurement.

Further, a pair-wise limit can be added for call admission control between communities. In this embodiment, for a given community link between two communities I and J, the following parameters may be defined:  $L_{IJ}$ , which represents the limit on a total bandwidth to be used by the community link IJ for the telephony application;  $M_{IJ}$ , which represents the threshold at which resource element selection is performed; and  $I_{IJ}$ , which represents the usage of transmission resources of the community link IJ. A community link does not have an N parameter since a link has no direction and the concept of incoming or outgoing calls does not apply. These limits may be dynamic and monitored in response to throughput measurements.

In some embodiments, for a terminal in community to establish a new call with a terminal in community J, the following must be satisfied:

$$\sum_{\text{all } K} T_{IK} < L_I \text{ and } T_{IJ} < L_{IJ}$$

The first clause essentially states that the traffic between community I and all other communities must be less than the threshold limit  $L_I$ . Alternatively, the comparison may be made to the threshold limit  $N_I$ . The second clause specifies that the traffic on the link IJ between communities I and J must be less than the threshold  $L_{IJ}$ . If either of the two clauses are not satisfied, then the call request from a terminal in community I is denied. A threshold  $M_{IJ}$  is also specified for the link IJ between communities I and J to specify a limit at which resource selection is performed.

The limits  $L_I$ ,  $L_{IJ}$ ,  $M_I$ , and  $M_{IJ}$  may be static (that is, they remain fixed) or adaptive (that is, they may change with changing conditions of the data network). For example, as the data network traffic increases, the threshold values may decrease. Furthermore, the threshold values may change in response to throughput measurements. The connection manager can collect statistics regarding the network (such as by accessing a network monitor or other node such as a router or gateway) to determine the conditions of the network. The connection manager can also collect statistics from the throughput measurement response signals sent between network resources. Based on the conditions, e.g., large delays or packet losses, the threshold values may be decreased to maintain high quality of service.

As illustrated in FIGURE 10, the first community 402 has the following parameters:  $T_1$ ,  $L_1$ ,  $M_1$ ,  $N_1$  the second community 404 has the following parameters:  $T_2$ ,  $L_2$ ,  $M_2$ , and  $N_2$  and the third community 406 has the following parameters:  $T_3$ ,  $L_3$ ,  $M_3$ , and  $N_3$ . The community link 432

has the following parameters:  $T_{12}$ ,  $L_{12}$ ,  $M_{12}$  and the community link 430 has the following parameters:  $T_{23}$ ,  $L_{23}$ , and  $M_{23}$ .

Referring to FIGURES 11A-11B, the call admissions control procedure is illustrated for a  
5 call between an originating terminal in one community (community I) and a destination terminal  
in a second community (community J). In the example of FIGURES 11A-11B, a first connection  
manager 500 services community I and a second connection manager 501 services community J.  
The connection manager 500 receives a CALL\_SETUP message from a terminal in community I  
that includes a list of supported audio codecs and a list of supported packet sizes. The connection  
10 manager 500 then determines (at 502) whether the call is an intra-community or an inter-  
community call. If the call is an intra-community call, then the connection manager 500 in  
community I performs intra-community call processing and exchanges messages between the  
terminals involved in the call session (at 504). Codec and other resource element selection may  
be performed as described above if the traffic  $T_I$  exceeds the threshold value  $M_I$ .

15 If the call is an inter-community call, then the connection manager 500 determines (at  
506) the name of the originating community, in this case community I. The connection manager  
500 then checks the attributes  $L_I$ ,  $M_I$ , and  $N_I$  of the community I. At this point, the connection  
manager 500 checks the traffic  $T_{IK}$  (between community I and all other communities) against the  
limit  $L_I$  (or  $N_I$ ). If  $T_{IK}$  exceeds  $L_I$  (or  $N_I$ ), or if the throughput measurement does not meet the  
20 throughput measurement requirement, then the call is denied by the connection manager 500.  
However, if the call request is allowed to proceed, then a candidate list of codecs and packet  
sizes is then created. Such a list of codecs and packet sizes may be further restricted based on the

values of the thresholds  $L_I$ ,  $M_I$ , and  $N_I$ . The bandwidth for community I is reserved to reserve capacity for the requested call. This allows the connection manager 500 to monitor the available bandwidth for further inter-community call requests in the community I.

A signal may be sent from connection manager 500 to the originating terminal 503 to initiate the sending of a throughput measurement (at 507) from the originating terminal 503 in community I to the destination terminal 505 in community J. The throughput measurement may comprise the trace route and monitoring combination as described in FIGURE 3. Alternatively, a trace route may originate from connection manager 500, propagate via the destination terminal 505, and the originating terminal 503, and then propagate back to the connection manager 500. Optionally, RSVP signaling, as used in FIGURE 5, may be used to reserve network resources on a transmission path to ensure a specific quality of service. It should be noted that step 507 may be optionally implemented at any stage in the call setup procedure.

A call request message is sent (at 508) from the connection manager 500 to the connection manager 501 that is assigned to community J. The message includes the name of the originating community I as well as the candidate list of codecs and packet sizes. In response to the message from the connection manager 500, the connection manager 501 determines the destination community name J, the community link IJ, and checks the limits  $L_J$ ,  $M_J$ , and  $N_J$ ,  $L_{IJ}$  and  $M_{IJ}$  (at 510). Such a check includes checking if value of  $T_{IJ}$  exceeds  $L_{IJ}$ . Also, the value of  $T_{JK}$  (total traffic of inter-community calls between community J and all other communities) is evaluated against  $L_J$  (or  $N_J$ ). If  $T_{JK}$  exceeds  $L_J$  (or  $N_J$ ) or  $T_{IJ}$  exceeds  $L_{IJ}$ , then the call is denied and the connection manager 501 informs the connection manager 500 of the call termination.

The connection manager 501 may also check  $T_{IJ}$  against  $M_{IJ}$  and  $T_{JK}$  (total traffic from community J) against  $M_J$ , to determine if resource selection is needed.

From the limits, the connection manager 501 may further restrict the list of allowed codecs and packet sizes. Bandwidth is then reserved for the community J and link IJ for the requested call. The connection manager 501 then sends a CALL\_SETUP message (at 512) to the destination terminal in community J. The Call Setup message includes the codec and packet size candidate list. In response to the CALL\_SETUP message, the destination terminal sends back a CALL\_CONNECT message (at 514) that identifies a selected codec and a packet size. The connection manager 501 and destination terminal may select a codec and packet size using techniques described in connection with FIGURES 3-5, which uses a ranking algorithm. Based on the returned CALL\_CONNECT message identifying the selected codec and packet size, the connection manager 501 corrects (at 516) the expected bandwidth usage of community I and link IJ. The connection manager 501 then sends back (at 518) a Connect/Answer message to the connection manager 500 that includes an identification of the community link (IJ) and the selected codec and packet size. Based on the identification of the selected codec and packet size, the expected bandwidth usage in the community I for the call session is corrected, and the expected bandwidth usage of the community link IJ is updated (at 520).

At this point, a call has been connected between the originating terminal and the destination terminal in communities I and J, respectively. If the originating terminal desires to terminate the phone call, then it sends a release message (at 522) to the connection manager 500. In response, the connection manager 501 updates (at 530) its bandwidth usage of community I and link IJ and sends a release message (at 524) to the connection manager 501. In response to

the release message, the connection manager 501 updates (at 526) the bandwidth usage for community J and link IJ to reflect termination of the call. The connection manager 501 sends a release complete message (at 528) to the destination connection manager to terminate the call.

A call management method and apparatus has been described to offer call admissions  
5 control and selection of resource elements to more effectively manage usage of a data network for telephony communications while providing a higher quality of service.

While the invention has been disclosed with respect to a limited number of embodiments,  
those skilled in the art will appreciate numerous modifications and variations therefrom. It is  
intended that the appended claims cover all such modifications and variations as fall within the  
10 true spirit and scope of the invention.